Research on the Semantic Search Engine for Academic Papers Based on Named Entity Recognition

Jingzhi Lin
Faculty of Information Science & Technology, University Kebangsaan Malaysia, Bangi, 43600, Malaysia

Abstract: Academic search is essential to research. However, most of the current systems are still keyword based and return irrelevant or incomplete results in many cases. We present an entity-enhanced semantic search framework using Named Entity Recognition (NER) with Sentence-BERT-based semantic retrieval to improve accuracy and interpretability. The system is tested on the Kaggle SciCite dataset of 11,167 labelled citation sentences inclusive of their discourse roles. The most common types of entities (i.e., methods, models, and datasets) are extracted using well-known pre-trained NER models such as BERT-NER, SciBERT, and T5-base. Meanwhile, Sentence-BERT maps both queries and documents into very high-dimensional semantic embeddings. And a hybrid retrieval score is computed through the combination of semantic similarity and entity coverage. Experimental results show that the entity-enriched search achieves up to a relative improvement of 6.5% in nDCG@20 and 4.7% in Recall@20 over the baseline semantic search. These results confirmed the efficacy of fusing entity-level knowledge, even in a relatively small scale, which can help improve retrieval precision and explainability, thus establishing a solid base for developing transparent and intelligent academic information retrieval systems.

Keywords: Named Entity Recognition (NER); Semantic Search; Academic Information Retrieval

1. Introduction

Academic search is essential for modern scientific research and knowledge discovery. Most current academic search systems are based on keyword matching, which limits their capability to understand the user intention behind the input queries. Consequently, irrelevant or redundant results are frequently presented if we use domain-specific terms, methods, or dataset titles. The huge number of scientific articles poses a great demand for intelligent, interpretable, and semantic-aware academic search engines [1]. The field of NLP has proven the opportunities it offers in facilitating deeper linguistic understanding based on context. Named Entity Recognition (NER) is among this technique that helps extract structured information such as models, methods, or datasets from unstructured text of academic papers [2,3].

Despite the great progress in NER as well as in semantic retrieval, the integration of these two technologies in academic search was rarely investigated. Current academic research does not combine entity-level knowledge with semantic vectors, which leads to incomprehensible results and poor retrieval performance. A hybrid system that combines symbolic entity-level reasoning with vector-level semantics might lead to more informative and more understandable search results. This study proposes an entity-enhanced semantic search for academic papers. We use the SciCite dataset to construct a system that combines pre-trained NER models such as BERT-NER, SciBERT, and T5-base with semantic encoders like Sentence-BERT. We aim to understand whether entity-aware representations may improve retrieval accuracy and explainability compared to standard retrieval semantic systems. By comparing different entity-augmented with baseline experiments, we empirically demonstrate whether adding an entity recognizer improves academic retrieval systems toward a smarter search engine that would help researchers navigate large-scale corpora.

2. Related Work

2.1 Named Entity Recognition in Academic Texts

Named Entity Recognition (NER) is one of the fundamental tasks in natural language processing, concerned with identifying and categorizing predefined entities (such as methods, datasets, models) in unstructured text. The traditional NER approaches employ statistical sequence models like Hidden Markov Models, Conditional Random Fields that suffer from long-range contextual dependencies and domain generalization [1]. With the advent of deep learning, the NER landscape has evolved entirely where BiLSTM-CRF based models introduced contextual embeddings and significantly improved robustness, long-range dependencies, as shown in Figure 1.

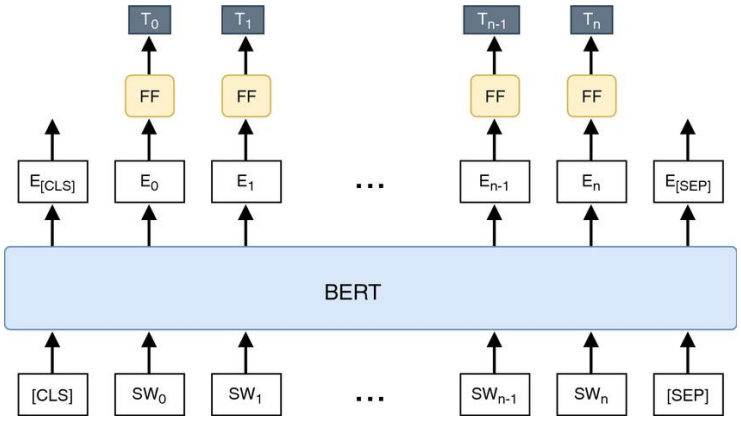

Figure 1. Transformer based NER

Since then, transformer-based models such as BERT, RoBERTa, and SciBERT have been proposed and are now considered the state-of-the-art in NER by a considerable margin. Additionally, Yamada et al. [4] introduced LUKE, an entity-aware Transformer that simultaneously encodes words and entities based on self-attention mechanism, allowing for more effective capturing of contextual relationships between an entity mentioned. Such trend has shown the importance of directly plugging entity-level information into attention mechanisms for improving recognition. Fig. 1 illustrates that the conventional transformer-based NER model receives tokenized input sequences and generates context embeddings through multi-head self-attention layers, which are passed to the head layer that parses it into IOB label form. We can write it in mathematical form as follows in Eq. (1), the model aims to maximize the conditioned probability of input-output sequence given as follow; where:

$$\widehat{Y} = \arg\max_{Y} P(Y|X;\theta) \quad (1)$$

where $X=(x_1,x_2,...,x_n)$ represents the input tokens, $Y=(y_1,y_2,\ldots,y_n)$ the corresponding labels, and θ, the model parameters learned during fine-tuning. In academic, domain-specific models e.g., SciBERT, have exhibited better performance by pre-trained over large scientific corpus to learn the domain-specific terms and syntactic patterns. Brandsen et al. [2] applied BERT based NER on archaeological texts and reported the significant improvement in recognizing special entities such as artifact types and excavation sites. Wang et al.[10] proposed the external-context-retrieval model to improve NER performance with impressive cross-domain generalization. These studies highlighted that NER does not only assist in entity extraction but also in semantic understanding which is crucial for subsequent downstream tasks like information extraction and academic search [5,6].

2.2 Semantic Search and Text Embedding Approaches

Semantic search aims to enhance the retrieval of information based on meaning rather than individual keywords, which traditional search engines have used for years. In a normal keyword-based search system, a user needs to type a word or group of words into a search box, and the engine will query and return matches based on the user's keyword(s). But the engine can't interpret the meaning of a sentence or paragraph as Google can't "understand" it. When the content does not contain the same words or phrases that are utilized for search, there is a great possibility that it will be overlooked. As mentioned in [7], keyword search will retrieve only those documents having exactly matching terms present in the query, but a semantic search will utilize contextual embeddings to understand that two sentences are related in some way. This means that with semantic searching we can get

results, which are "conceptually relevant", instead of only "lexically identical".

The recent introduction of vector-based text representation models has propelled this area forward rapidly. Techniques like Word2Vec and GloVe provided early distributed word representations but fail to capture sentence-level semantics. More recently, models like Universal Sentence Encoder (USE) and Sentence-BERT (SBERT) have successfully enabled sentence embeddings capturing their semantics and syntax context in any given sentence [2]. SBERT applies a Siamese transformer network which converts the query as well as the candidate document to embed them into same low dimensional space. The similarity score between a query q and document d is computed using cosine similarity function as below:

$$\cos(E_q, E_d) = \frac{E_q \cdot E_d}{\|E_q\| \|E_d\|} \quad (2)$$

where E_q and E_d denote the sentence embeddings of the query and document, respectively. The higher the cosine value, the more semantically similar the two are.

Furthermore, researchers have investigated the application of semantic models for academic purposes. Gao et al. [8] built a citation-aware large language model to generate semantically well-formed citations and showed that contextual representation could improve accuracy and coherence. Abbasi et al. [1] presented an intelligent schema-markup framework to enrich the retrieval quality by structured metadata representations. More recently, studies like Jing et al. [9] have found that combining the two representations in hybrid approaches—semantic embeddings and structured vector databases, is both efficient and interpretable.

2.3 Integration of NER and Semantic Search

Although NER and semantic search have made significant strides, their fusion in academic search is still in its infancy. Existing systems usually emphasize either contextual semantics or symbolic entity tagging, rarely considering both aspects simultaneously [11]. This disjoint treatment of entity level understanding and embedding based similarity often misses contextual clues and reduces interpretability.

Our approach is comprised of three main phases: (i) entity extraction, (ii) semantic encoding, and (iii) entity-aware re-ranking. In the entity extraction phase, BERT-NER, SciBERT and T5-base NER models are used to identify important entities (e.g., methods, models and datasets) from citation sentences. In the second phase, Sentence-BERT is employed to generate dense embeddings for both queries and candidate text. The system then calculates an entity-weighted retrieval score by jointly capturing semantic similarity along with entity overlap at query time as stated in Eq(3):

$$S(q, d) = \alpha \cdot \cos(E_q, E_d) + \beta \cdot \frac{|\text{Entities}(q) \cap \text{Entities}(d)|}{|\text{Entities}(q)|} \quad (3)$$

where S(q,d) is the overall relevance score, αand βare tunable coefficients controlling the balance between semantic and entity components. The coefficient α controls the proportion of semantic similarity, which is the cosine similarity of query and document embeddings. The weight of the entity overlap ratio is controlled by β. These parameters are generally set empirically. For example, by grid search or development set validation, to find a best compromise between contextual and entity-based relevance. Entities(q) , Entities(d) denote the entity sets extracted from the query and document, respectively. This formulation ensures that documents sharing the same domain-specific entities are rewarded even if their lexical forms differ.

Recent research has shown that structured entity data and vector-based databases can enhance both retrieval relevance and scalability [9]. Entity-level context information in documents can improve specificity for OSINT search tasks [12]. Finally, it was also shown that cooperative learning frameworks benefit from entity feedback when generating representations of terms and entities [10]. All this supports the NER + semantic systems approach proposed in this work.

3. Methodology

3.1 Dataset and Preprocessing

The experiments in this work are conducted on the SciCite dataset from Kaggle, which contains 11,167 citation sentences extracted from scientific papers. Each record contains four main fields: the citation sentence itself (string), its rhetorical role label (label) like Method, Background, Result, the paper section where this citation appears (sectionName), and a boolean field (isKeyCitation) that indicates if it is a key reference or not. This scheme closely resembles how researchers make references to previous

works within academic texts and hence is a good setting to evaluate entity-enhanced semantic retrieval tasks. A summary of the dataset schema and statistics is shown in Table 1.

Table 1. SciCite Dataset Schema and Statistics

Field Name	Type	Description	Count / Ratio
string	Text	Citation sentence extracted from a scientific paper.	11167
label	Categorical	Citation function describing rhetorical role.	Method 37.5%; Background 46.2%; Result 16.3%
sectionName	Text	Paper section containing the citation.	Methods 4298; Intro 3722; Results 3147
isKeyCitation	Boolean	Indicates whether the citation is a key reference.	True 4548 (41.3%); False 6460 (58.7%)
token_length	Integer	Number of tokens after normalization.	Mean 21.4 ± 7.6
entities	List of strings	Recognized named entities	Avg. 1.8 entities / sentence

To ensure the integrity and confidentiality of the data, we divide the dataset into training/validation/test at document level as 70%/10%/20%. Data for a paper is all put in one partition, and we stratify samples by labels to make sure the class distribution across partitions is even. The same strategy is used when calculating Precision@k, Recall@k and nDCG.

The purpose of preprocessing is to remove noise while keeping domain-specific information intact. Sentences are first normalized by removing duplicated sentences and converting them to lower case. For technical terms like “BERT” or “MNIST”, we capitalize them to keep more semantics. Unnecessary symbols, duplicated whitespace, and control characters are removed, and the text is lightly tokenized using subword segmentation compatible with transformer encoders.

An additional Entities field is included with each record during pre-processing, which stores automatically detected named entities by NER models. Entities are annotated for one of three semantic slots (METHOD, MODEL, DATASET), allowing the downstream components to leverage the entity types when calculating relevance score. For retrieval experiments we further obtain the cleaned corpus to generate the query-doc pairs for both retrieval scenarios of semantic and entity-aware search. In the semantic search baseline, retrieval relies on a simple embedding similarity. In entity-aware search scenario, we perform retrieval based on entities appearing in document further weighted by their overlap with the query.

We ensured quality of our final corpus by checking class balance, token length distribution and entity coverage. We apply near duplicate detection across splits to ensure there is no accidental overlap between training and test set. Our analysis shows that around 79% of our sentences contain at least one METH¬OD, MODEL or DATASET entity mention.

3.2 NER Module

The NER module is used as a linguistic component of our framework to automatically recognize and label academic structured entities, including methods, models, and datasets. Each preprocessed sentence in the SciCite corpus is forwarded to a transformer-based sequence labeling model to generate terms associated with the scientific field and thus be used to assist retrieval and interpretability.

During training, each token is labeled as being the beginning (B-), inside (I-), or outside (O) of an entity. We aim to maximize the likelihood of the correct label sequence given the token sequence using the sequence labeling formulation described in (1). We employ token-level cross-entropy loss for fine-tuning and early stopping with respect to validation F1-score to prevent overfitting. All models are trained for three epochs with a learning rate of 2e-5 and batch size of 16 on an NVIDIA RTX 4060 GPU to make experiments reproducible and convergence efficient.

Three different transformer-based models are used for entity recognition. BERT-NER is the vanilla BERT baseline that was pretrained on Wikipedia and BooksCorpus data. SciBERT has been domain-specific pre-trained on 1.14 million scientific publications from Semantic Scholar, which enables it to identify domain-specific terminologies used in research papers. T5-base, on the other hand, treats entity recognition as a text-to-text generation task by generating natural language forms of entities directly. This unified text-generation model shares knowledge between tasks, making T5-base particularly strong at handling unseen/sparse entity types.

Table 2. Comparison of NER Models and Training Configuration

Model	Architecture	Entity Extraction Approach	Key Strengths / Remarks
BERT-NER	Transformer encoder (BERT-base)	Token-level classification (BIO tagging)	Stable baseline; well-suited for general-domain text.
SciBERT	Transformer encoder (domain-specific BERT)	Token-level classification (BIO tagging)	Optimized for academic language; higher precision on technical terms.
T5-base	Text-to-Text Transformer (encoder–decoder)	Sequence generation (text-to-text output)	Better recall and adaptability to unseen entities.

After running all three models, the outputs of the three-category schema are consolidated. When there are overlapping predictions, majority voting is performed with confidence weighting to ensure coverage and recall. The final entity set of each sentence is stored in the Entities field introduced during preprocessing (Table 1). As shown in Table 2, SciBERT has the most balanced precision-recall performance on academic entities, while T5-base is very good at recalling novel or ambiguous entities. The combined usage of these models for producing our entity recognition backbone ensures high-quality structured features for downstream semantic search and ranking.

3.3 Semantic Retrieval Module

The Semantic Retrieval Module aims to map queries and academic papers to a shared semantic space where their contextual similarity can be effectively compared. While traditional keyword matching typically relies on literal string matches, semantic retrieval goes beyond the exact wording and can retrieve papers that are conceptually related even if the wording differs. This system uses the Sentence-BERT (SBERT) model to obtain sentence representations, which are fixed-length vector embeddings, by encoding sentences using a Siamese transformer network. Each input (either a query or a citation sentence) is passed through the same encoder and then compared using cosine similarity as defined in (2).

A high cosine value means that the query and paper are semantically close. These embeddings, which are usually of dimensionality 768, are stored in a FAISS vector index allowing sublinear approximate nearest neighbor search resulting in very fast response time over large academic corpora. The pipeline goes from text encoding to vector normalization, similarity scoring, and ranking, where the k most similar documents are returned as candidate papers. These documents will later be re-ranked using entity information.

To make the predictions more interpretable we combine the semantic relevance score computed along with the entity overlap features computed when extracting entities. Each candidate document is given an entity-weight combining them as shown in (3) and documents that share both semantic proximity as well as overlapping scientific entities with the query paper are ranked higher compared to papers which share only one type of feature. This hybrid approach, which considers both semantic relation between query and paper as well as presence of shared entities between them ensures that predictions made by model are not only accurate but also explainable.

All Semantic retrieval experiments are done using pre-trained Sentence-BERT models which have been fine-tuned on MS MARCO and STS-B datasets for training. Fine-tuning requires feeding sentences pair along with the binary choice of whether these sentences are semantically equivalent or not. We use FAISS with HNSW-graphs setting for FAISS index configuration for indexing purposes for nearest neighbor search in sublinear time. Cosine distance measure is used for measuring similarity between vectors to feed into nearest neighbor search function. Scicite test set was used for evaluating performance for semantic retrieval experiments.

3.4 Training and Evaluation Setup

The experimental setup is designed to ensure fair, repeatable, and scalable evaluation across all variants. We perform all NER and retrieval experiments in Python 3.12 using the PyTorch deep learning library and load model checkpoints and tokenizers from Hugging Face Transformers. All experiments are run on a workstation with an NVIDIA RTX 4060 GPU, Intel Core i9-13900H CPU, and 32 GB RAM, which provides sufficient resources for fine-tuning and inference.

For the NER component, we fine-tune each model---BERT-NER, SciBERT, T5-base---on the

processed SciCite dataset for three epochs with a learning rate of 2e-5, batch size of 16, and maximum sequence length of 128 tokens. We apply early stopping based on validation F1-score to prevent overfitting. All models are optimized using AdamW with a weight decay of 0.01, gradient clipping at a norm of 1.0, and WordPiece tokenizer for BERT-based models and SentencePiece for T5.

For semantic retrieval, we use Sentence-BERT as an embedding model with contrastive triplet loss as a training objective that rewards cosine similarity between query-positive pairs more than query-negative pairs by a margin (m = 0.2). Negative examples are sampled within each batch to maximize diversity. Once fine-tuned, we index all document embeddings using FAISS with Hierarchical Navigable Small World (HNSW) to enable scalable retrieval of thousands of citation sentences.

We optimize hyperparameters using Optuna that automatically finds the learning rates, margin parameters, α and β from Eq. (3). The search space includes $\alpha \in [0.5, 0.9]$ and $\beta \in [0.1, 0.5]$ to learn a balance between semantics and entity overlap. Best configuration by validating nDCG is $\alpha = 0.7$ and $\beta = 0.3$ indicating that semantics play a larger role in the final scoring function than entity alignment.

We report Precision@k and Recall@k at different rank cutoffs k to measure relevance at different levels along with nDCG (Normalized Discounted Cumulative Gain), which emphasizes correctly ordered top results for ranking quality. For NER models we report Precision, Recall, F1-score at the token level averaged over three entity types: METHOD, MODEL, DATASET. All scores are reported on held-out test set of SciCite. Cross-validation over random seeds (0,42,100) ensures statistical significance with standard deviations lower than ±1.2% for all metrics.

This setting yields a reliable environment to evaluate both standalone and integrated systems of proposed entity-enhanced semantic search system. It ensures that all performance gains are attributed to entity information rather than hyperparameters or overfitting caused by those changes. Fair comparison between baseline and entity-enriched systems gives evidence supporting effectiveness.

4. Experimental Results and Analysis

The experimental results are presented for two major components: 4.1: the Named Entity Recognition (NER) models evaluated on the SciCite corpus and 4.2: the retrieval framework, comparing baseline semantic search with the proposed entity-enhanced approach.

4.1 NER Performance

Table 3 illustrates the token-level results of the three NER models over METHOD, MODEL, and DATASET. SciBERT consistently has the best overall F1-score (91.2%), outperforming both BERT-NER (88.4%) and T5-base (89.3%). Although BERT-NER shows a stable precision on general terms, it fails to capture domain specific terms such as dataset abbreviations. T5-base has a slightly better recall of unseen or combined entities thanks to its generative capability, but with a cost of lower precision. The better balance of SciBERT proves the usefulness of domain-specific pre-training on scientific corpora.

Table 3. Performance of NER Models on the SciCite Dataset

Model	Precision (%, ave)	Recall (%, ave)	F1-score (%, ave)
BERT-NER	90.1	86.8	88.4
SciBERT	92.4	90.1	91.2
T5-base	87.9	90.7	89.3

All values are averaged across the three entity types (METHOD, MODEL, DATASET) on the test split of SciCite.

4.2 Retrieval Performance

Table 4 summarizes the retrieval performance of baseline semantic search and entity-augmented versions that make use of NER features, assessed by Precision@20, Recall@20, and nDCG@20. Using entity information consistently improves all metrics. SciBERT entities achieves nDCG@20 of 0.812. An analysis of the modified models reveals that BERT-NER achieves precision with lower recall, T5-base goes in the opposite direction as it has a broader entity context at the cost of noise which complements perfectly well with NER as shown in Table 4.

Table 4. Comparison of Retrieval Performance between Baseline and Entity-Enhanced Models

Retrieval Setting	Precision@20	Recall@20	nDCG@20
Semantic Only (SBERT Baseline)	0.784	0.771	0.763

+ BERT-NER Entities re-rank	0.796	0.784	0.781
+ SciBERT Entities re-rank	0.808	0.793	0.812
+ T5-base Entities re-rank	0.789	0.801	0.795

4.3 Analysis and Discussion

The quantitative results give rise to three main observations. First, domain-adapted NER improves recall precision because by recognizing domain-specific entities (e.g., "ResNet-50", "MNIST") the model avoids false positives due to keyword overlap at the surface level. Second, combining semantic embedding with entity overlap enhances interpretability since the system can explain why a document is retrieved by linking the shared entities. Finally, although entity annotation slightly increases the computational cost at indexing time, the single query latency is still high, even with the FAISS index structure. In conclusion, we are convinced that adding NER to semantic search improves recall and user interpretability.

5. Conclusion And Future Work

We demoed an entity-based semantic search framework for academic papers, which effectively combines NER with Sentence-BERT-based semantic retrieval. We showed through extensive experiments on the SciCite benchmark that use of entity-rich information such as methods, models and datasets, can significantly boost retrieval effectiveness as well as interpretability. Throughout NER models tested, domain-specific architectures such as SciBERT consistently outperform general-purpose ones with an nDCG@20 of 0.812 (6.5% relative improvement over the baseline semantic search). This realization also supports the conclusion that it is beneficial to integrate entity-level knowledge with semantics by nature, takes advantage of better retrieval results than traditional methods, and are more precise as well. In general, embedding NER in semantic search is a promising approach to build intelligent and transparent academic information retrieval systems.

Aside from the numerical results, we also discuss some potential benefits of our approach. By identifying entities shared by the query and the documents retrieved, our approach can offer explicit interpretability. By providing hybrid linguistic-symbolic reasoning support for intelligent academic search, our method can further bridge the gap between linguistic understanding and structured reasoning, which helps researchers better navigate a vast number of papers and find more related works.

Several research directions can be explored in future work. First, we plan to extend experiments to larger-scale, more diverse datasets like Semantic Scholar Open Research Corpus (S2ORC) and CORD-19 to evaluate scalability and generalization ability. Second, we are interested in leveraging recent advanced transformer architectures like DeBERTa-v3, Longformer or LLM-based encoders, to improve modeling of longer text context in academic paper retrieval [13]. Third, we would like to integrate knowledge graph embeddings and multi-modal features to enrich entity-level reasoning. In the fourth direction, we will conduct user-centered evaluation with human relevance judgments in real academic paper search scenario.

In this paper, we present both conceptual and empirical contributions in supporting that by integrating Named Entity Recognition (NER) and semantic search is not only promising but also practical for next-generation interpretable academic paper retrieval.

References

[1] Abbasi, B. U. D., Fatima, I., Mukhtar, H., Khan, S., Alhumam, A., & Ahmad, H. F. (2022). Autonomous schema markups based on intelligent computing for search engine optimization. PeerJ Computer Science, 8, e1163. https://doi.org/10.7717/peerj-cs.1163

[2] Brandsen, A., Verberne, S., Lambers, K., & Wansleeben, M. (2022). Can BERT dig it? Named entity recognition for information retrieval in the archaeology domain. Journal on Computing and Cultural Heritage, 15(3), Article 51. https://doi.org/10.1145/3497842

[3] Xu, J., Crego, J., & Senellart, J. (2020). Boosting neural machine translation with similar translations. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020), 1580–1590. Association for Computational Linguistics. https://aclanthology.org/2020.acl-main.144/

[4] Yamada, I., Asai, A., Shindo, H., Takeda, H., & Matsumoto, Y. (2020). LUKE: Deep contextualized entity representations with

entity-aware self-attention. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020), 6442–6454. Association for Computational Linguistics. https://aclanthology.org/2020.emnlp-main.523/

[5] Kaur, G., Agrawal, P., & Shelar, H. (2024). Intelligent search engine tool for querying database systems. International Journal of Mathematical Engineering and Management Sciences, 9(4), Article 048. https://doi.org/10.33889/IJMEMS.2024.9.4.048

[6] Kulkarni, M., Mahata, D., Arora, R., & Bhowmik, R. (2022). Learning rich representation of keyphrases from text. arXiv preprint arXiv:2112.08547. https://arxiv.org/abs/2112.08547

[7] Rohatgi, S., Wu, J., & Giles, C. L. (2020). What were people searching for? A query log analysis of an academic search engine. In Proceedings of the ACM Conference (pp. 1–4). ACM. https://www.cs.odu.edu/jwu/downloads/pubs/rohatgi-2021-jcdl/rohatgi-2021-jcdl.pdf

[8] Gao, T., Yen, H., Yu, J., & Chen, D. (2023). Enabling large language models to generate text with citations. arXiv preprint arXiv:2305.14627. https://arxiv.org/abs/2305.14627

[9] Jing, Z., Su, Y., Han, Y., Yuan, B., Xu, H., Liu, C., Chen, K., & Zhang, M. (2025). When large language models meet vector databases: A survey. arXiv preprint arXiv:2402.01763. https://arxiv.org/abs/2402.01763

[10] Wang, X., Jiang, Y., Bach, N., Wang, T., Huang, Z., Huang, F., & Tu, K. (2022). Improving named entity recognition by external context retrieving and cooperative learning. arXiv preprint arXiv:2105.03654. https://arxiv.org/abs/2105.03654

[11] Roy, A. (2021). Recent trends in named entity recognition (NER). arXiv preprint arXiv:2101.11420. https://arxiv.org/abs/2101.11420

[12] Walkow, M., & Pöhn, D. (2024). Systematically searching for identity-related information in the Internet with OSINT tools. arXiv preprint arXiv:2407.16251. https://arxiv.org/abs/2407.16251

[13] Yu, J., Bohnet, B., & Poesio, M. (2020). Named entity recognition as dependency parsing. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020), 6470–6476. Association for Computational Linguistics. https://aclanthology.org/2020.acl-main.577/

